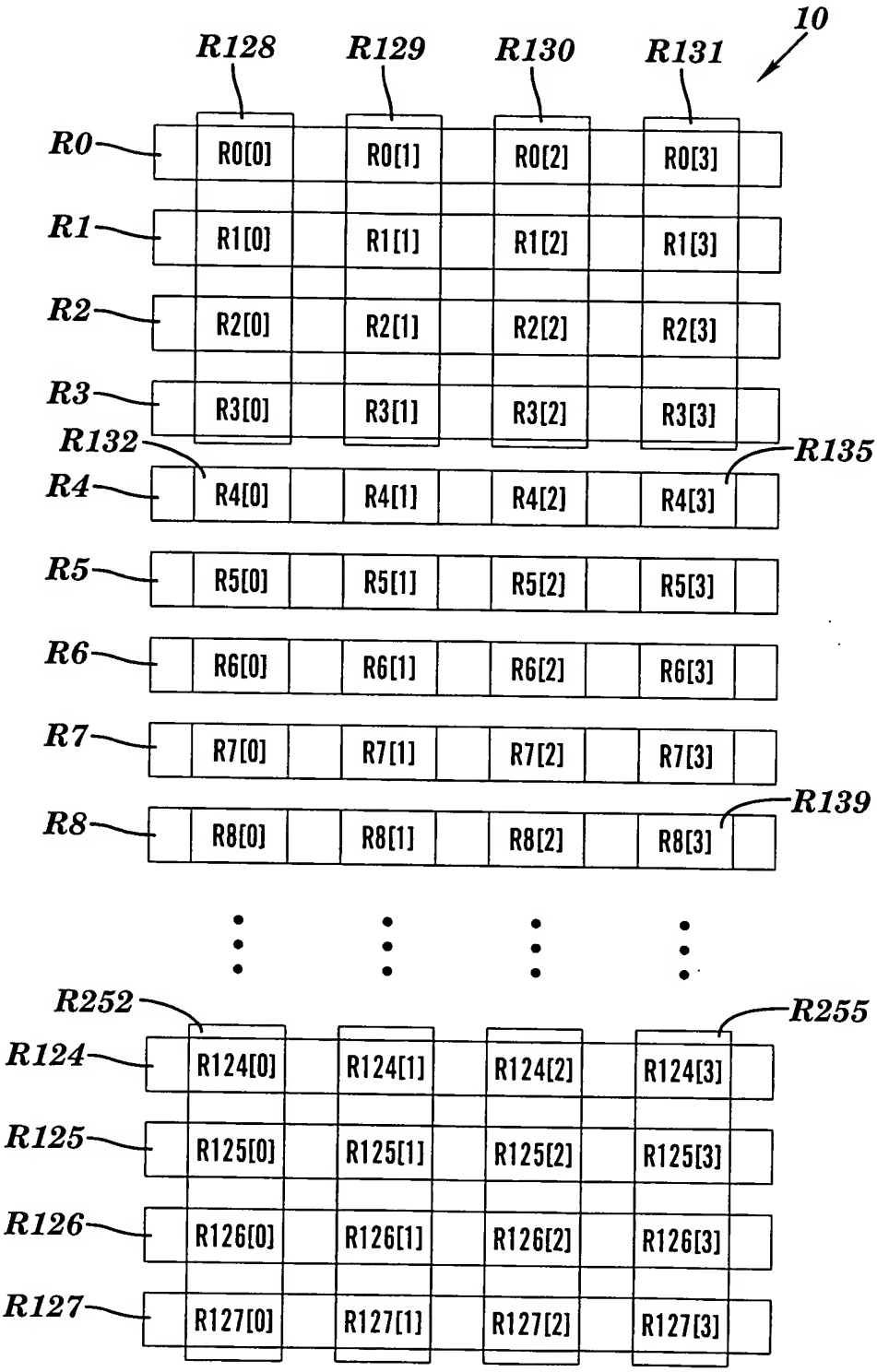
10
R128
R129
R130
R131
R0
R1
R2
R3
R132
R135
R4
R5
R6
R7
R8
R139
R252
R255
R124
R125
R126
R127
R0[0]
R0[1]
R0[2]
R0[3]
R1[0]
R1[1]
R1[2]
R1[3]
R2[0]
R2[1]
R2[2]
R2[3]
R3[0]
R3[1]
R3[2]
R3[3]
R4[0]
R4[1]
R4[2]
R4[3]
R5[0]
R5[1]
R5[2]
R5[3]
R6[0]
R6[1]
R6[2]
R6[3]
R7[0]
R7[1]
R7[2]
R7[3]
R8[0]
R8[1]
R8[2]
R8[3]
R124[0]
R124[1]
R124[2]
R124[3]
R125[0]
R125[1]
R125[2]
R125[3]
R126[0]
R126[1]
R126[2]
R126[3]
R127[0]
R127[1]
R127[2]
R127[3]

FIG. 1

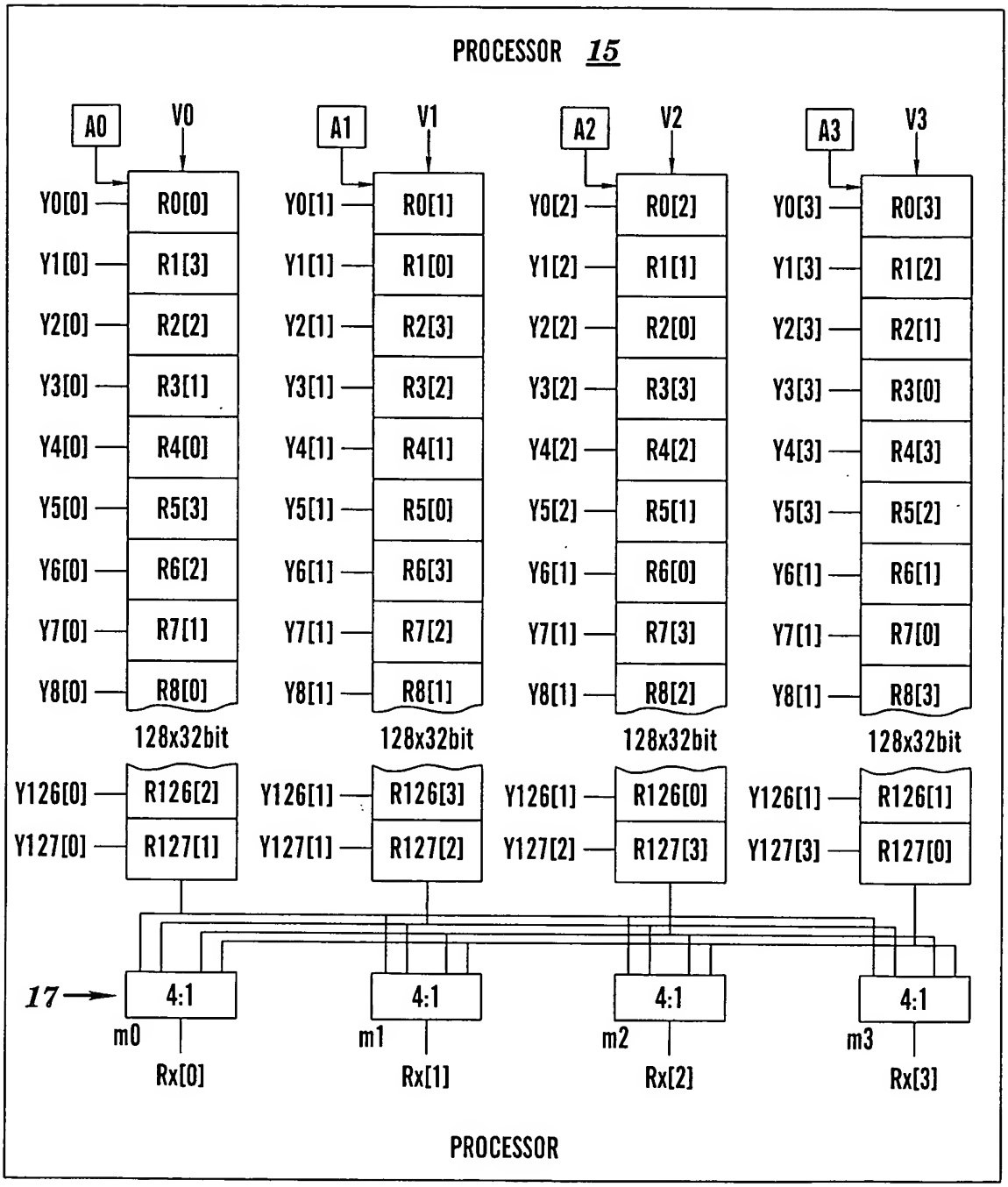
PROCESSOR 15
A0
V0
A1
V1
A2
V2
A3
V3
Y0[0]
R0[0]
Y0[1]
R0[1]
Y0[2]
R0[2]
Y0[3]
R0[3]
Y1[0]
R1[3]
Y1[1]
R1[0]
Y1[2]
R1[1]
Y1[3]
R1[2]
Y2[0]
R2[2]
Y2[1]
R2[3]
Y2[2]
R2[0]
Y2[3]
R2[1]
Y3[0]
R3[1]
Y3[1]
R3[2]
Y3[2]
R3[3]
Y3[3]
R3[0]
Y4[0]
R4[0]
Y4[1]
R4[1]
Y4[2]
R4[2]
Y4[3]
R4[3]
Y5[0]
R5[3]
Y5[1]
R5[0]
Y5[2]
R5[1]
Y5[3]
R5[2]
Y6[0]
R6[2]
Y6[1]
R6[3]
Y6[1]
R6[0]
Y6[1]
R6[1]
Y7[0]
R7[1]
Y7[1]
R7[2]
Y7[1]
R7[3]
Y7[1]
R7[0]
Y8[0]
R8[0]
Y8[1]
R8[1]
Y8[1]
R8[2]
Y8[1]
R8[3]
128x32bit
128x32bit
128x32bit
128x32bit
Y126[0]
R126[2]
Y126[1]
R126[3]
Y126[1]
R126[0]
Y126[1]
R126[1]
Y127[0]
R127[1]
Y127[1]
R127[2]
Y127[2]
R127[3]
Y127[3]
R127[0]
17
4:1
4:1
4:1
4:1
m0
m1
m2
m3
Rx[0]
Rx[1]
Rx[2]
Rx[3]
PROCESSOR

FIG. 2

20

21	22	23	24	25	26	27	28	29
REG Rx	A0	A1	A2	A3	m0	m1	m2	m3
R0	0	0	0	0	0	1	2	3
R1	1	1	1	1	1	2	3	0
R2	2	2	2	2	2	3	0	1
R3	3	3	3	3	3	0	1	2
R4	4	4	4	4	0	1	2	3
R5	5	5	5	5	1	2	3	0
R6	6	6	6	6	2	3	0	1
⋮								
R126	126	126	126	126	2	3	0	1
R127	127	127	127	127	3	0	1	2
R128	0	1	2	3	0	1	2	3
R129	3	0	1	2	1	2	3	0
R130	2	3	0	1	2	3	0	1
R131	1	2	3	0	3	0	1	2
R132	4	5	6	7	0	1	2	3
R133	7	4	5	6	1	2	3	0
⋮								
R254	126	127	124	125	2	3	0	1
R255	125	126	127	124	3	0	1	2

FIG. 3

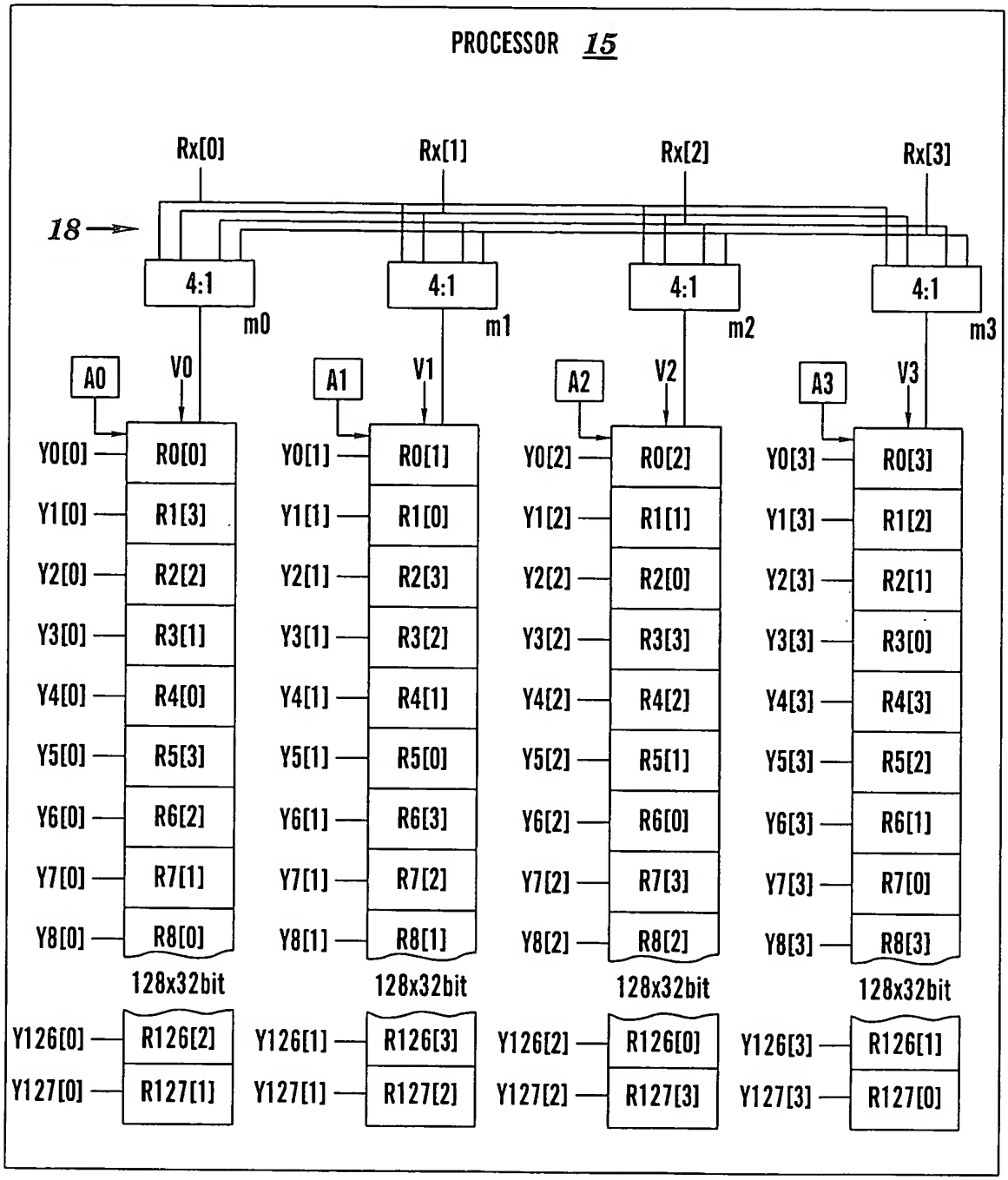
PROCESSOR 15
Rx[0]
Rx[1]
Rx[2]
Rx[3]
18
4:1
m0
m1
m2
m3
A0
V0
A1
V1
A2
V2
A3
V3
Y0[0]
R0[0]
Y1[0]
R1[3]
Y2[0]
R2[2]
Y3[0]
R3[1]
Y4[0]
R4[0]
Y5[0]
R5[3]
Y6[0]
R6[2]
Y7[0]
R7[1]
Y8[0]
R8[0]
128x32bit
Y126[0]
R126[2]
Y127[0]
R127[1]
Y0[1]
R0[1]
Y1[1]
R1[0]
Y2[1]
R2[3]
Y3[1]
R3[2]
Y4[1]
R4[1]
Y5[1]
R5[0]
Y6[1]
R6[3]
Y7[1]
R7[2]
Y8[1]
R8[1]
128x32bit
Y126[1]
R126[3]
Y127[1]
R127[2]
Y0[2]
R0[2]
Y1[2]
R1[1]
Y2[2]
R2[0]
Y3[2]
R3[3]
Y4[2]
R4[2]
Y5[2]
R5[1]
Y6[2]
R6[0]
Y7[2]
R7[3]
Y8[2]
R8[2]
128x32bit
Y126[2]
R126[0]
Y127[2]
R127[3]
Y0[3]
R0[3]
Y1[3]
R1[2]
Y2[3]
R2[1]
Y3[3]
R3[0]
Y4[3]
R4[3]
Y5[3]
R5[2]
Y6[3]
R6[1]
Y7[3]
R7[0]
Y8[3]
R8[3]
128x32bit
Y126[3]
R126[1]
Y127[3]
R127[0]

FIG. 4

40

41 REG Rx	42 A0	43 A1	44 A2	45 A3	46 m0	47 m1	48 m2	49 m3
R0	0	0	0	0	0	1	2	3
R1	1	1	1	1	3	0	1	2
R2	2	2	2	2	2	3	0	1
R3	3	3	3	3	1	2	3	0
R4	4	4	4	4	0	1	2	3
R5	5	5	5	5	3	0	1	2
R6	6	6	6	6	2	3	0	1
⋮								
R126	126	126	126	126	2	3	0	1
R127	127	127	127	127	1	2	3	0
R128	0	1	2	3	0	1	2	3
R129	3	0	1	2	3	0	1	2
R130	2	3	0	1	2	3	0	1
R131	1	2	3	0	1	2	3	0
R132	4	5	6	7	0	1	2	3
R133	7	4	5	6	3	0	1	2
⋮								
R254	126	127	124	125	2	3	0	1
R255	125	126	127	124	1	2	3	0

FIG. 5

REGISTER MOVE/COPY WITH SOURCE SELECTION: rDEST← rRA[aa,bb,cc,dd]

OPCODE	DEST	RA	aa	bb	cc	dd
8 BITS	8 BITS	8 BITS	2 BITS	2 BITS	2 BITS	2 BITS

FIG. 6A

REGISTER MERGE UNDER MASK: rDEST ← (rDEST, rRA)

OPCODE	DEST	RA	reserved	mask
8 BITS	8 BITS	8 BITS	4 BITS	4 BITS

FIG. 6B

SCALAR/VECTOR 3 OPERAND FP INSTRUCTION: rDEST← f(rRB, rRA [aa,aa,aa,aa])

OPCODE	aa	DEST	RA	RB
6 BITS	2 BITS	8 BITS	8 BITS	8 BITS

FIG. 6C

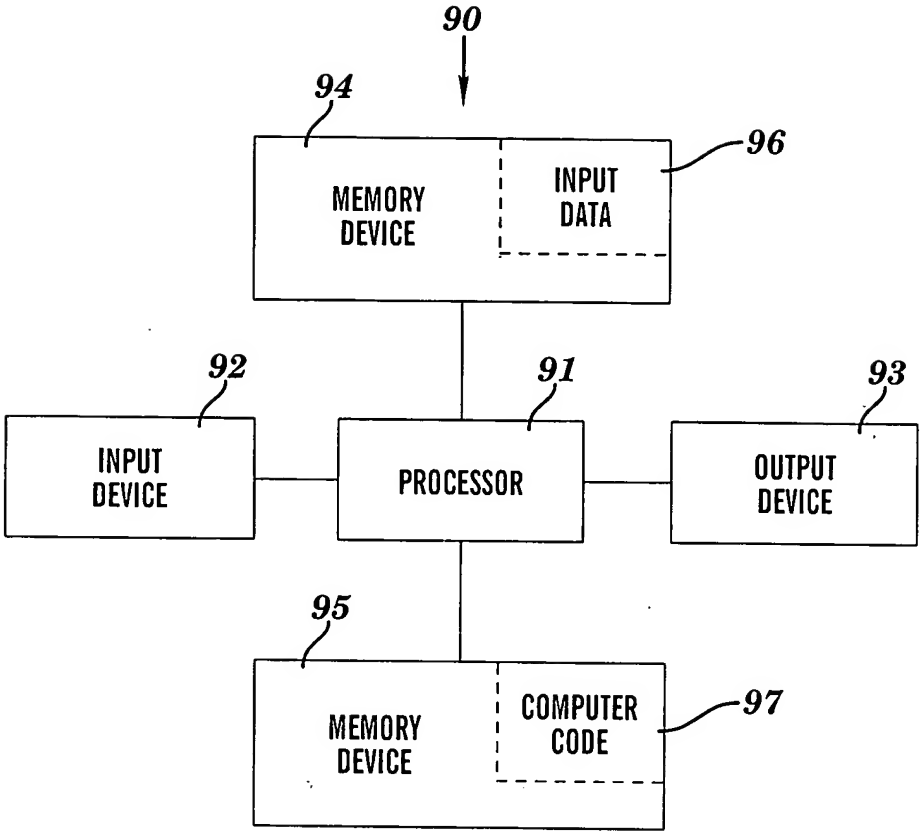
90
94
96
MEMORY DEVICE
INPUT DATA
92
91
93
INPUT DEVICE
PROCESSOR
OUTPUT DEVICE
95
97
MEMORY DEVICE
COMPUTER CODE

FIG. 7